FIG. 1

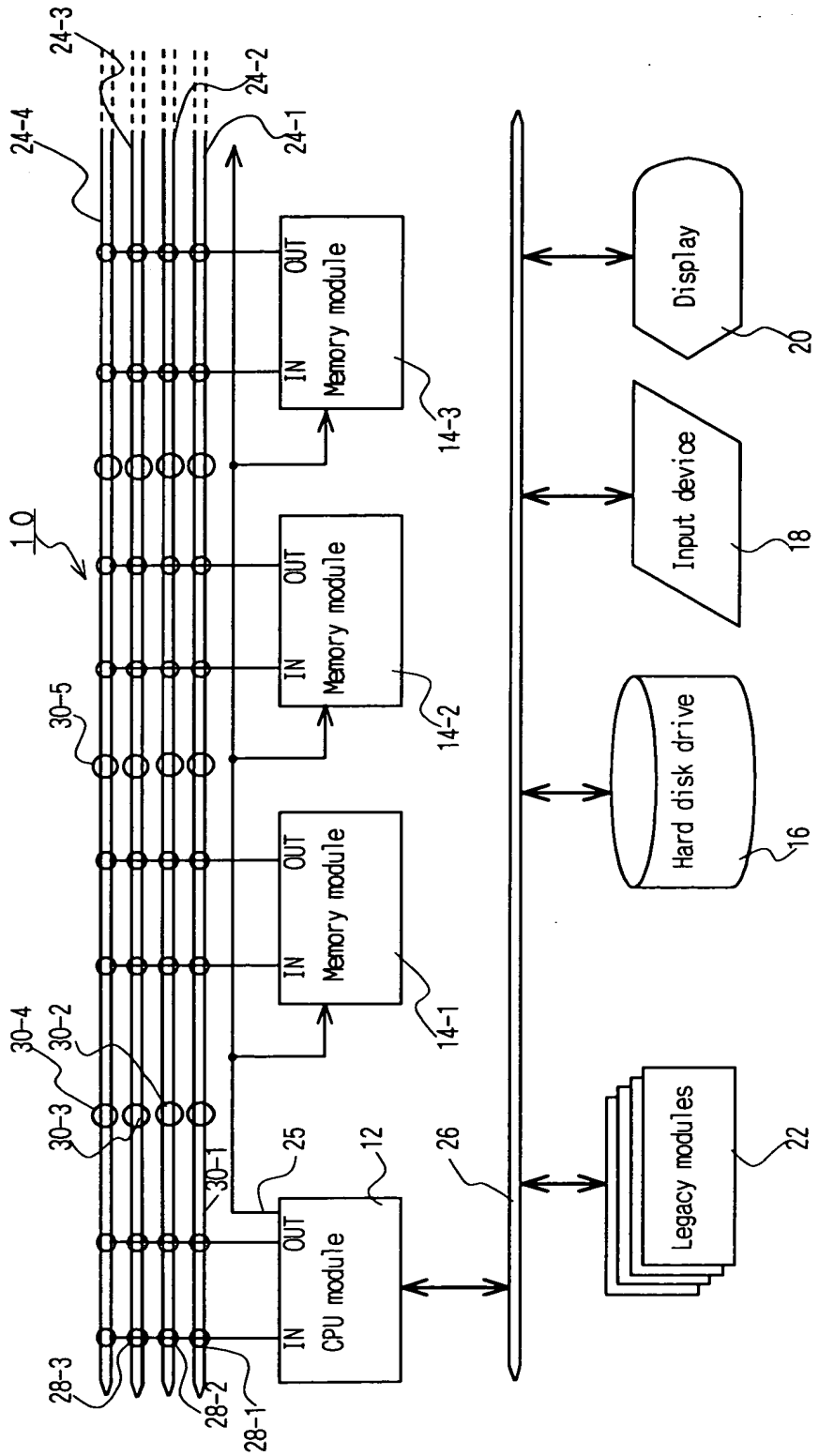

10
24-3
24-2
24-1
24-4
IN
OUT
Memory module
14-3
14-2
14-1
30-5
30-4
30-2
30-3
30-1
25
12
CPU module
26
28-3
28-2
28-1
Display
20
Input device
18
Hard disk drive
16
Legacy modules
22

FIG. 2

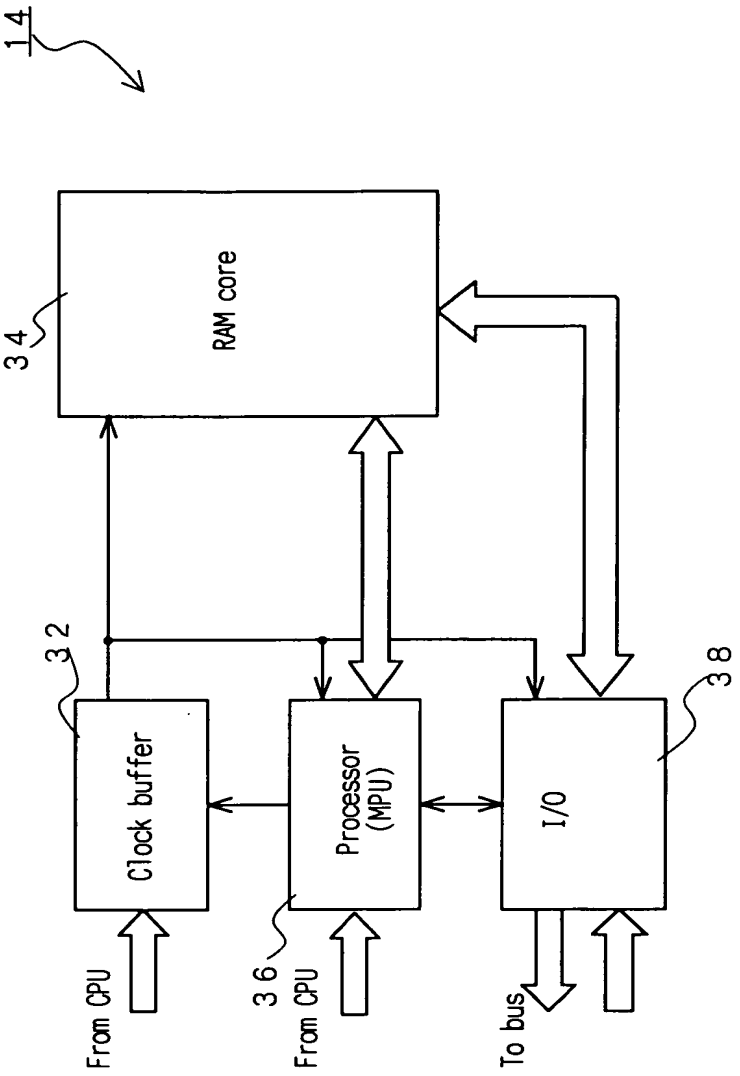

14
RAM core
34
32
38
Clock buffer
Processor (MPU)
I/O
From CPU
36
From CPU
To bus

FIG. 3

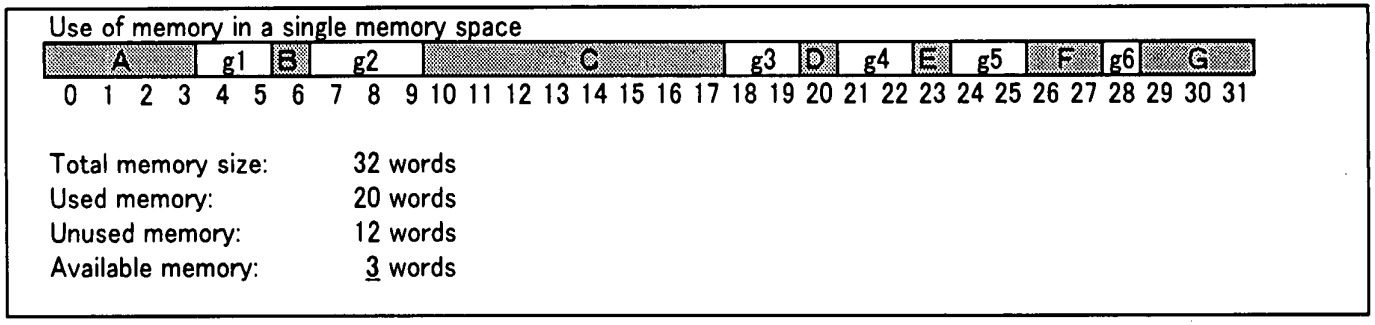


FIG. 4

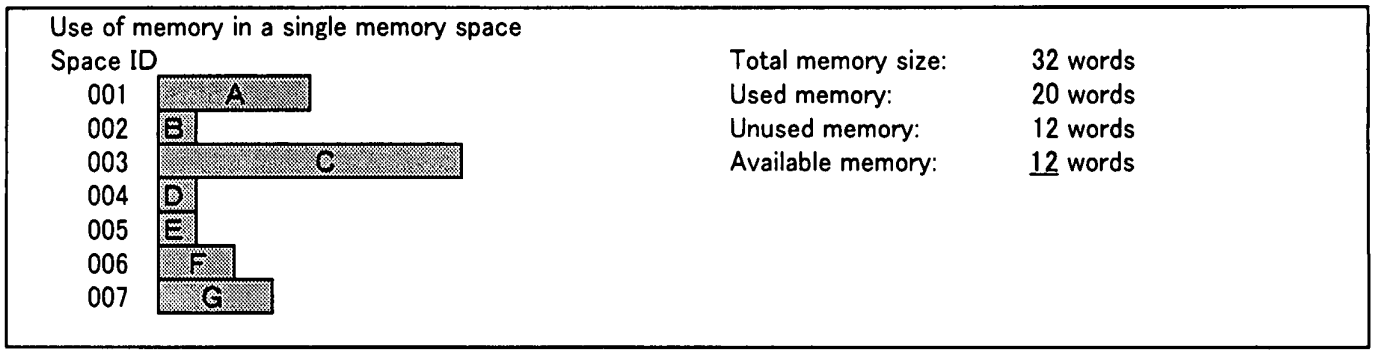


FIG. 5

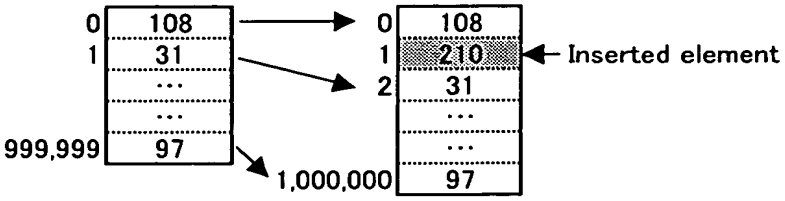

0
1
999,999
108
31
97
0
1
2
1,000,000
108
210
31
97
Inserted element

FIG. 6

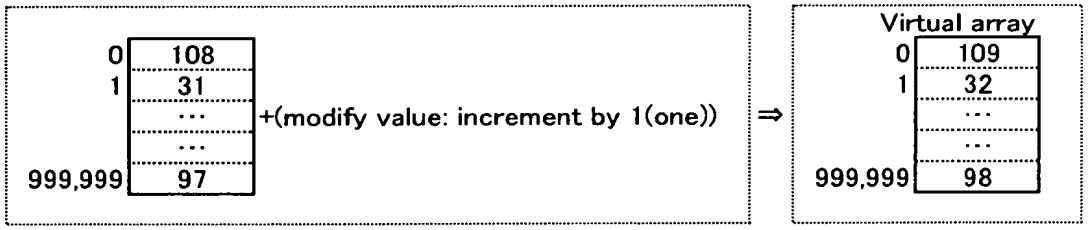

0
1
999,999
108
31
97
+(modify value: increment by 1(one))
⇒
Virtual array
0
1
999,999
109
32
98

FIG. 7

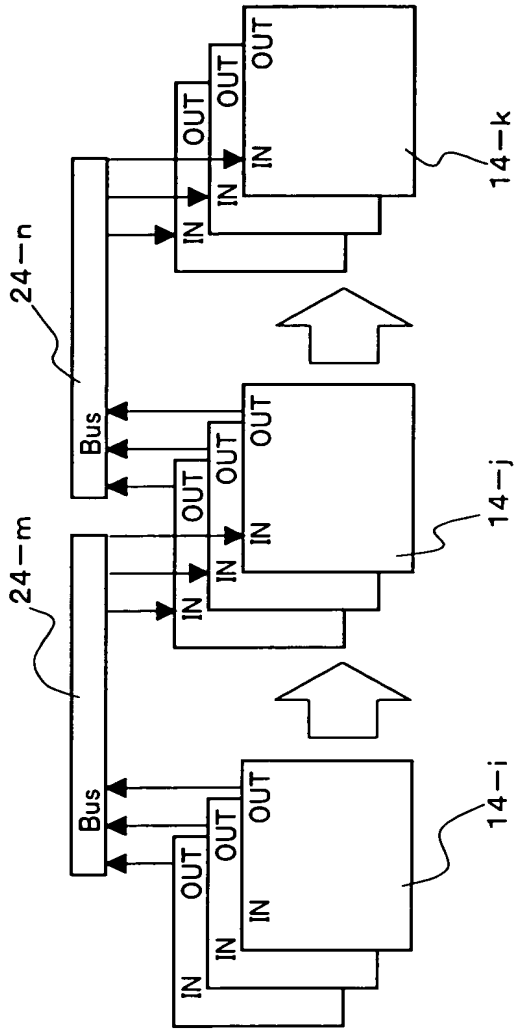


FIG. 8A

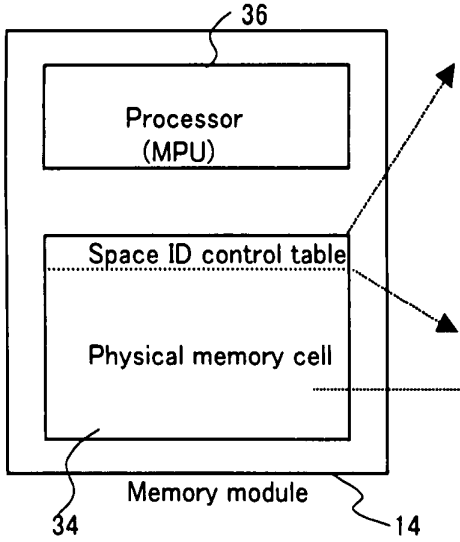


FIG. 8B

Space ID	Logical start address	Allocated region size	On-chip physical start address	Total size	Access control flags
010	100	60	0	164	R
010	160	4	80	164	R
016	0	20	60	20	RW
…	…	…	…	…	…
…	…	…	…	…	…
212	0	100	360	100	RW
–	–	–	–	–	–
–	–	–	–	–	–

Access control flag legend:
R: Read only
W: Write only
RW: Read/Write enabled

FIG. 8C

Content of physical memory cell

Physical address	ID	Logical address	
0	10	010:0100	
1	21	010:0101	Space ID 010
…			
59	33	010:0159	
60	2323	016:0000	Space ID 016
…			
79	3241	016:0019	
80	15	010:0159	Space ID 010
…			
83	8	010:0163	
…			
…			
360	-589	212:0000	Space ID 212
…			
459	-1022	212:0099	
…			
…			

FIG. 9A

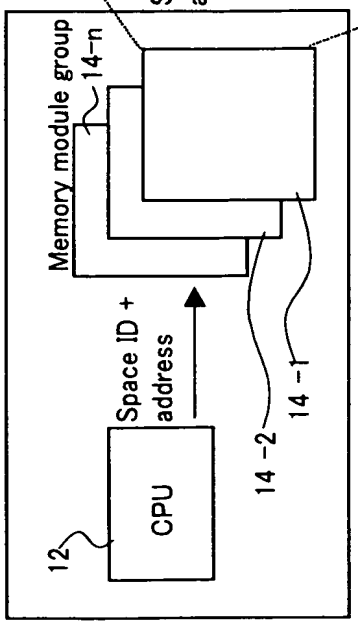


FIG. 9B

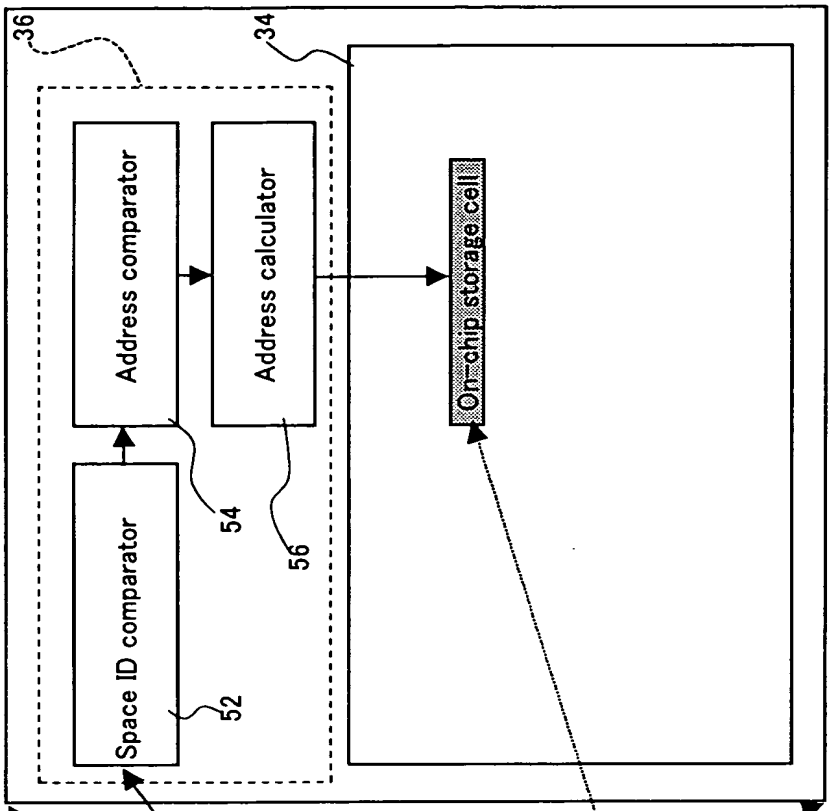


FIG. 9C

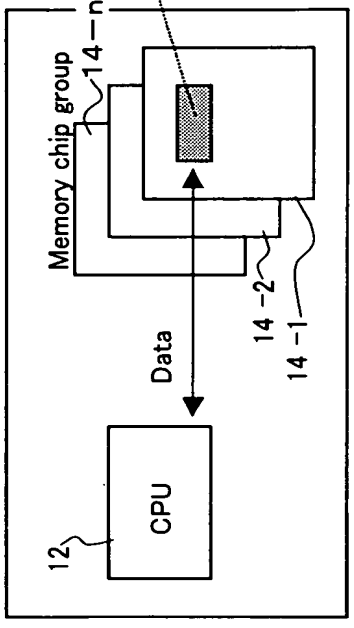
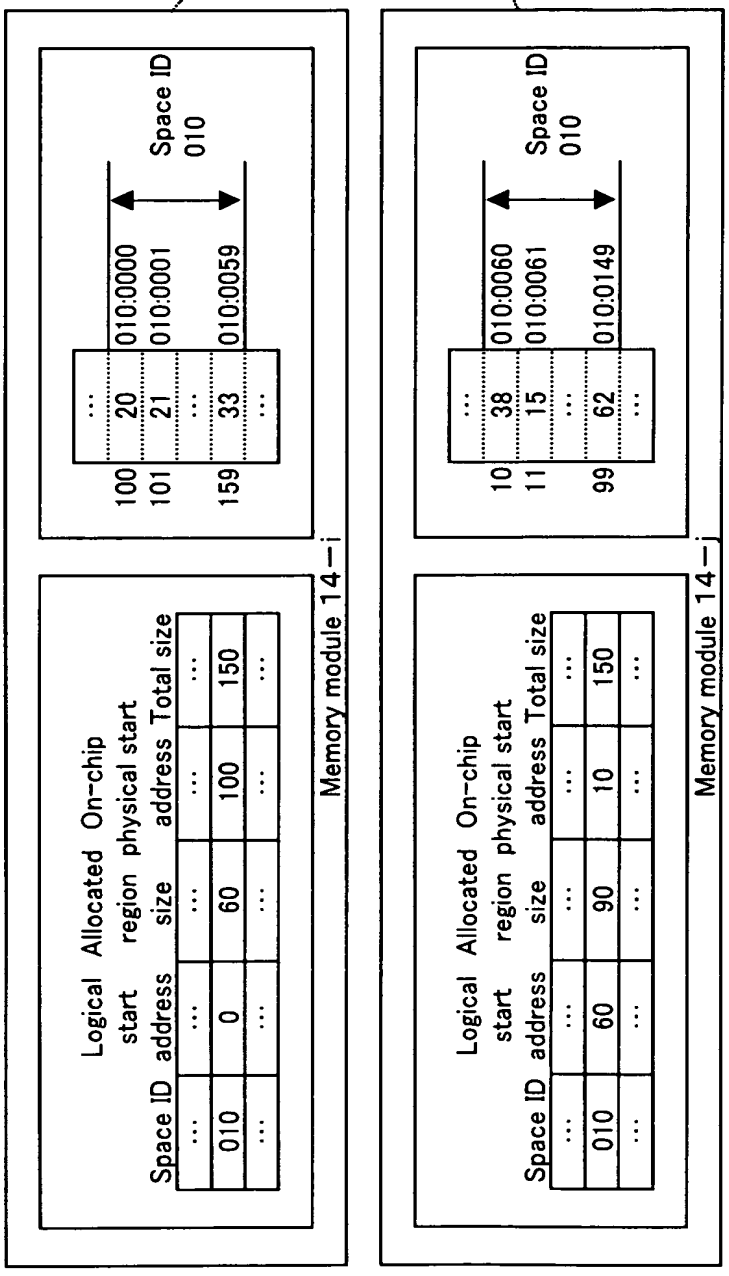


FIG. 10A



Space ID	Logical start address	Allocated region size	On-chip physical start address	Total size
...	...	...	...	...
010	0	60	100	150
...	...	...	...	...

Space ID	Logical start address	Allocated region size	On-chip physical start address	Total size
...	...	...	...	...
010	60	90	10	150
...	...	...	...	...

FIG. 10B

FIG. 10C

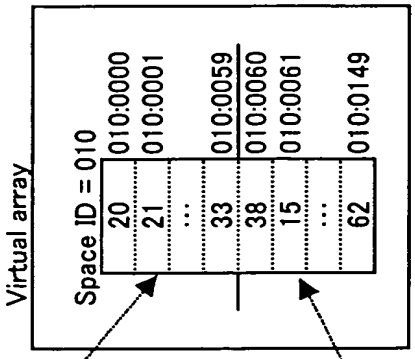


FIG. 11

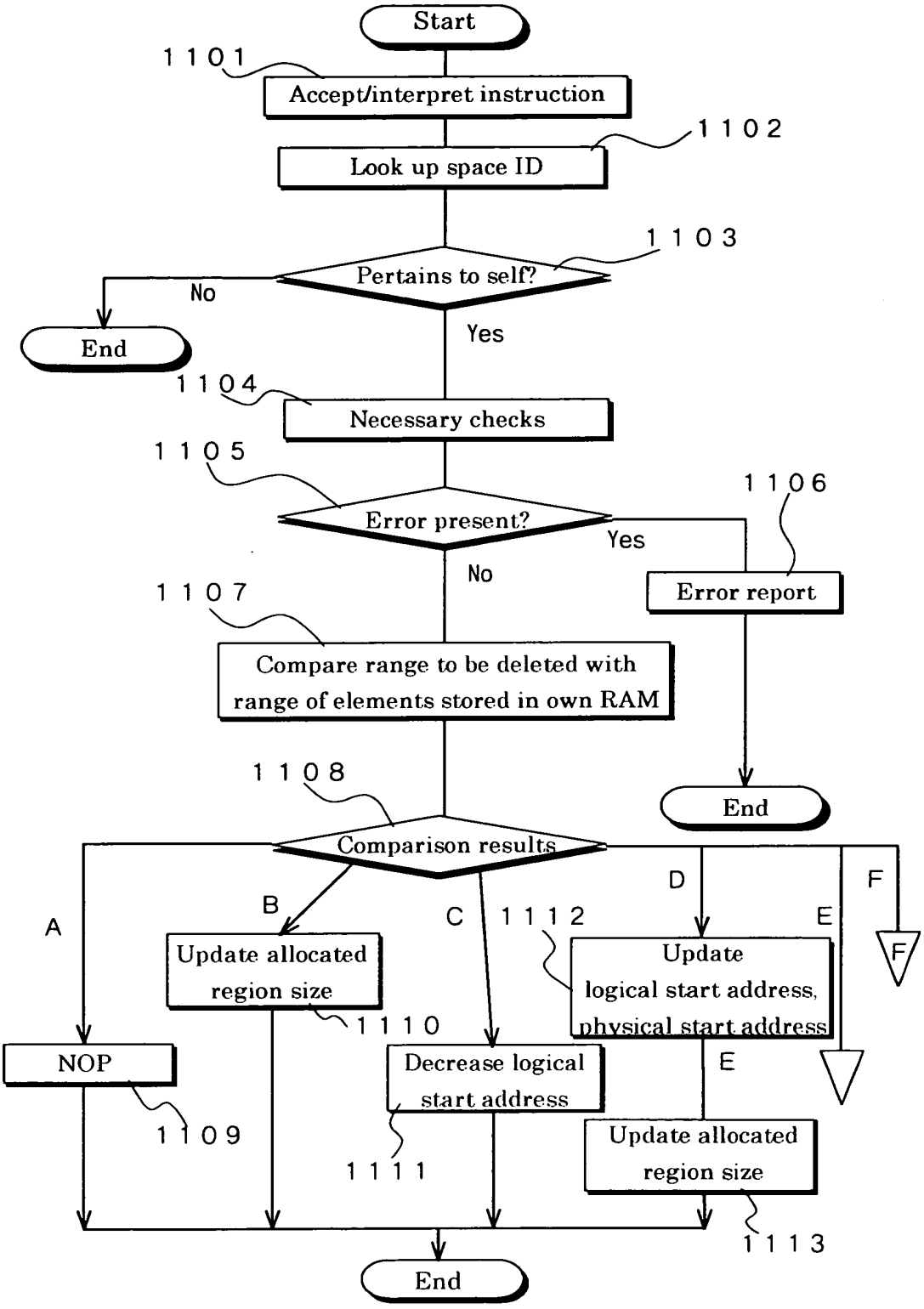

Start
1101
Accept/interpret instruction
1102
Look up space ID
1103
Pertains to self?
No
End
Yes
1104
Necessary checks
1105
Error present?
1106
Yes
No
Error report
1107
Compare range to be deleted with range of elements stored in own RAM
End
1108
Comparison results
A
B
C
D
E
F
1112
Update allocated region size
1110
Update logical start address, physical start address
NOP
1109
Decrease logical start address
1111
E
F
Update allocated region size
1113
End

FIG. 12A

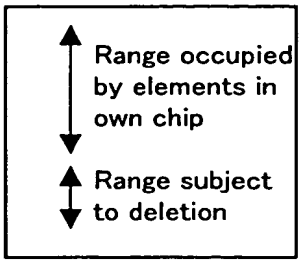


FIG. 12B

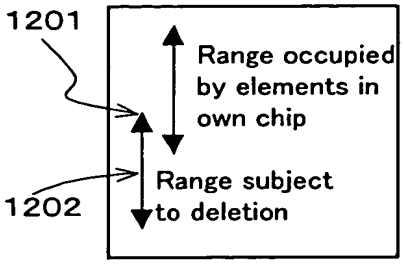


FIG. 12E

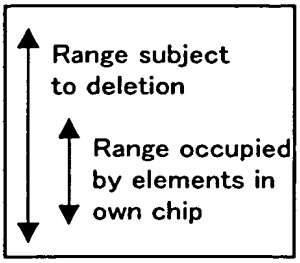


FIG. 12C

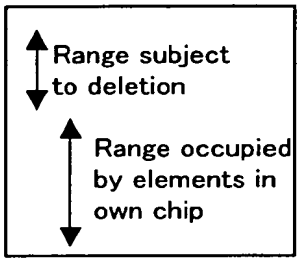


FIG. 12F

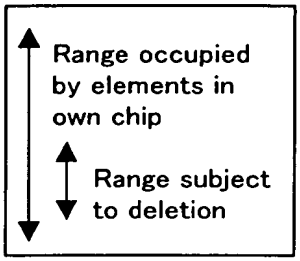


FIG. 12D

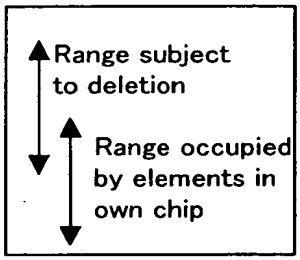


FIG. 13

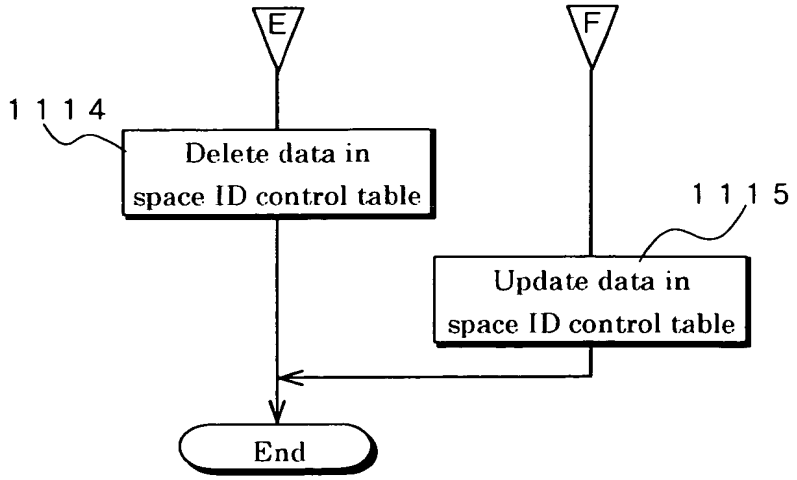


FIG. 14

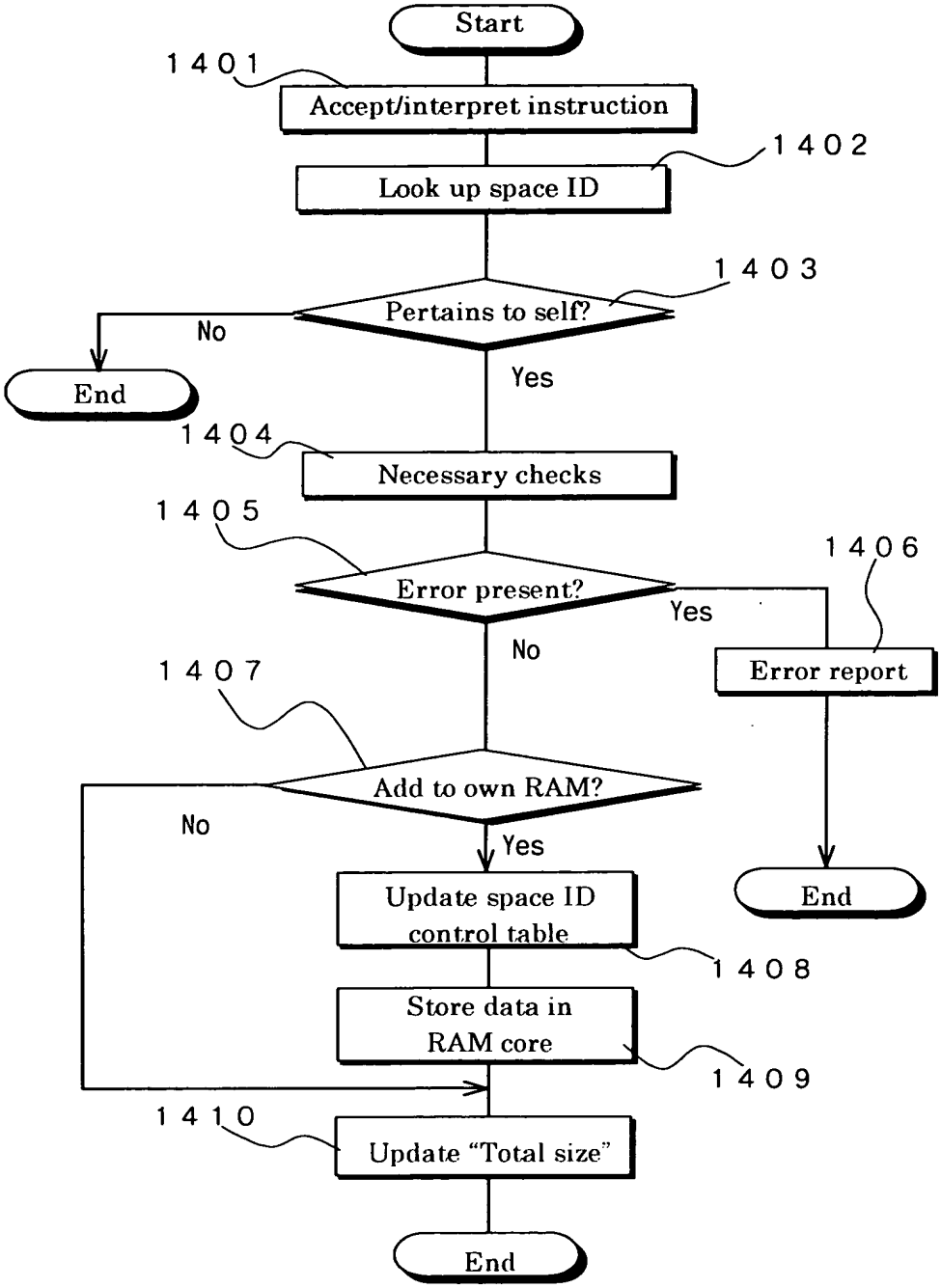

Start
1401
Accept/interpret instruction
1402
Look up space ID
1403
Pertains to self?
No
End
Yes
1404
Necessary checks
1405
Error present?
1406
Yes
Error report
No
1407
Add to own RAM?
No
Yes
End
Update space ID control table
1408
Store data in RAM core
1409
1410
Update "Total size"
End

FIG. 15A

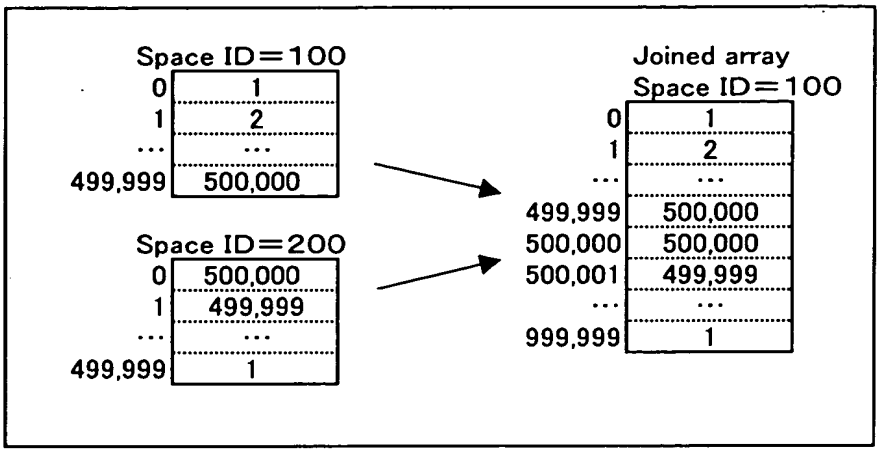


FIG. 15B

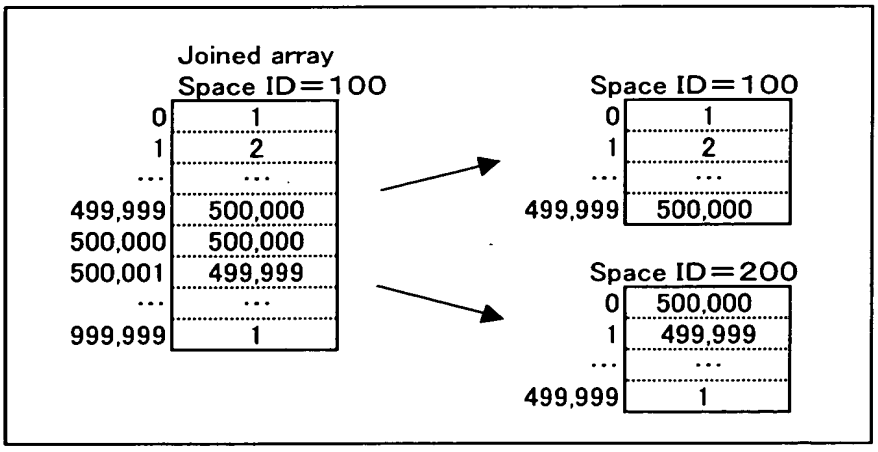


FIG. 16A

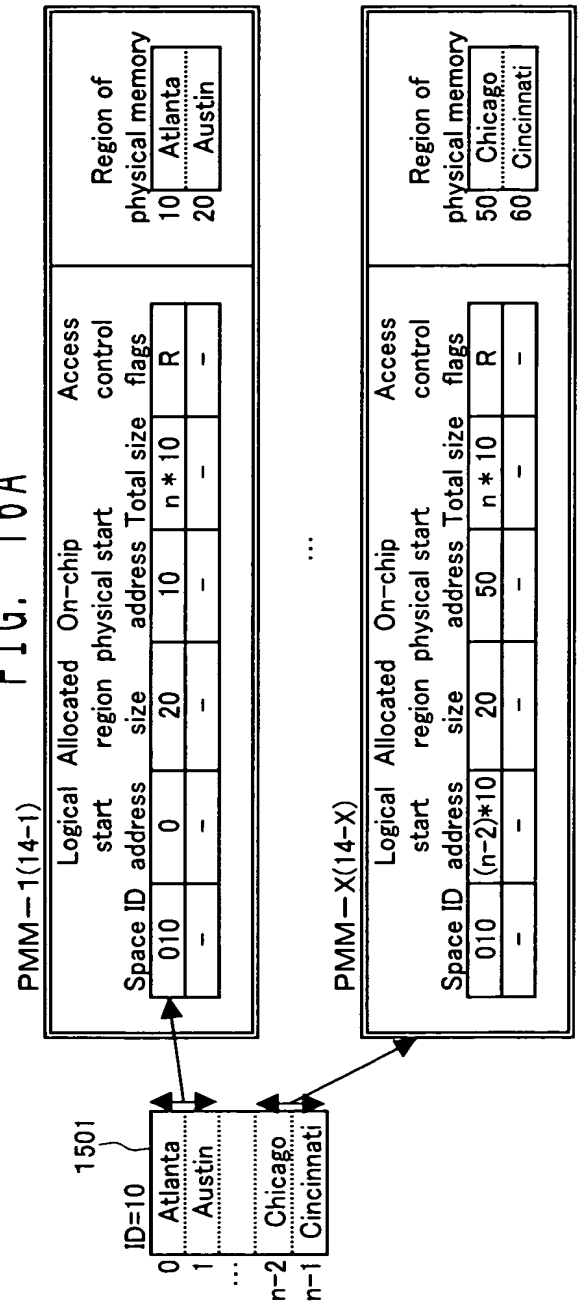


FIG. 16B

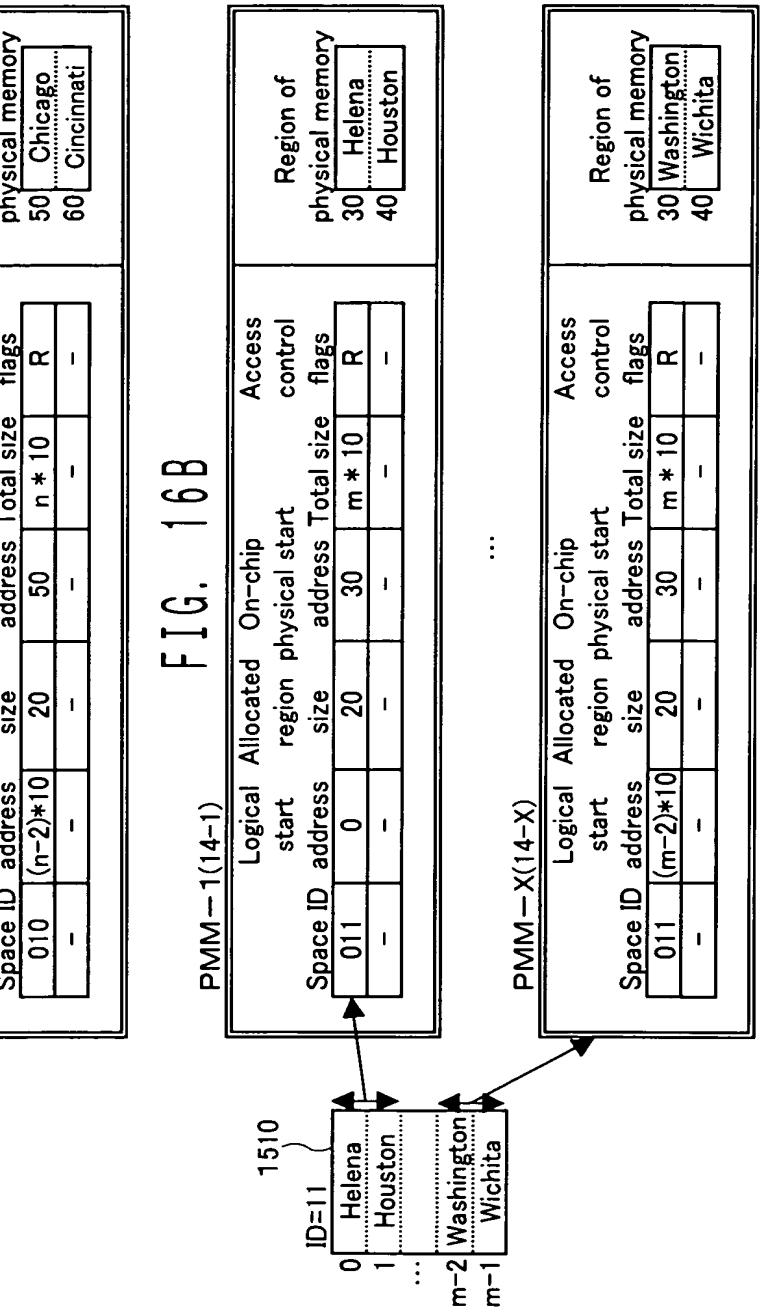


FIG. 17

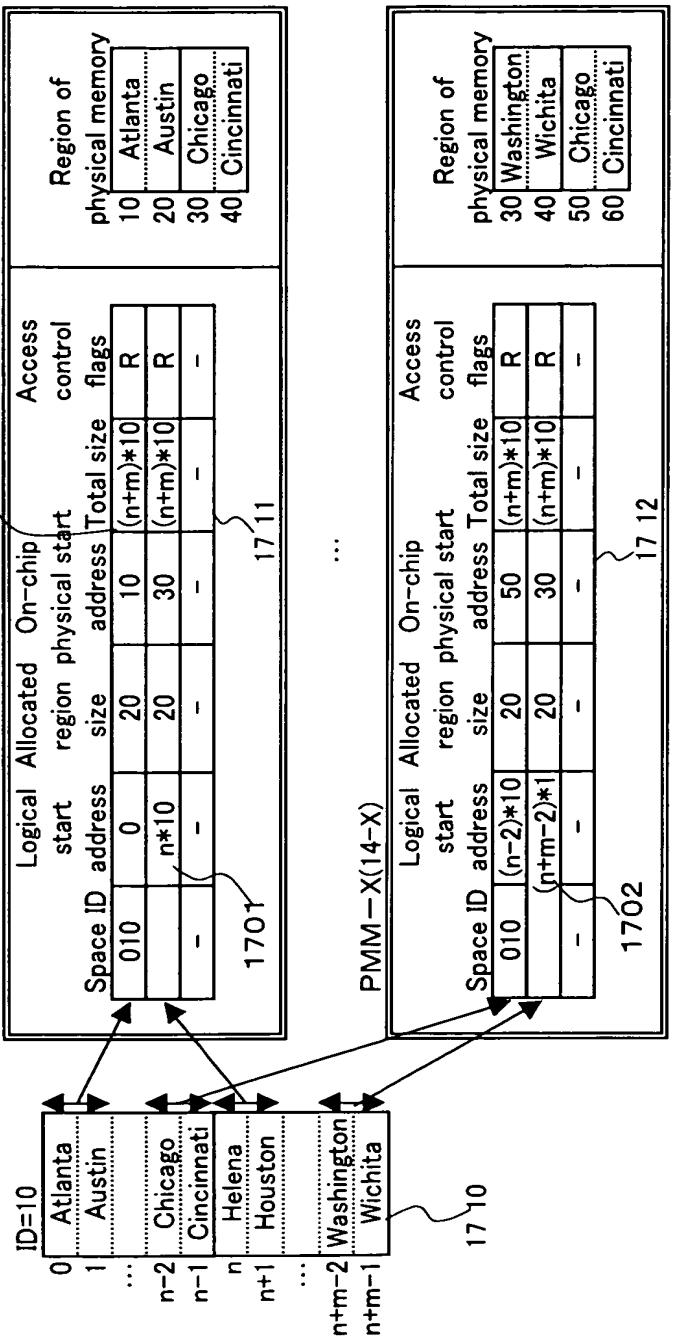

1703
PMM−1(14-1)
Logical start address
Allocated region size
On-chip physical start address
Total size
Access control flags
Space ID
010
n*10
Region of physical memory
10 Atlanta
20 Austin
30 Chicago
40 Cincinnati
1701
1711
PMM−X(14-X)
(n-2)*10
(n+m-2)*1
50
30
(n+m)*10
R
30 Washington
40 Wichita
50 Chicago
60 Cincinnati
1702
1712
ID=10
0 Atlanta
1 Austin
n-2 Chicago
n-1 Cincinnati
n Helena
n+1 Houston
n+m-2 Washington
n+m-1 Wichita
1710

FIG. 18

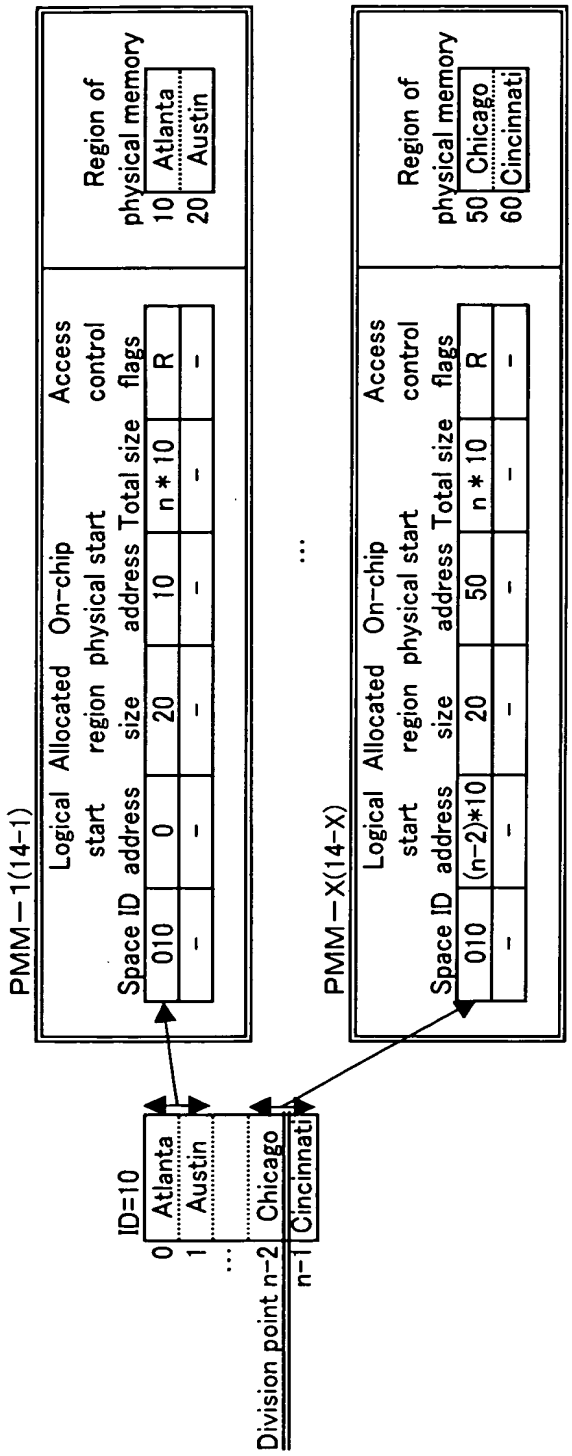

PMM−1(14-1)
Space ID
Logical start address
Allocated region size
On-chip physical start address
Total size
Access control flags
Region of physical memory
010
0
20
10
n * 10
R
10 Atlanta
20 Austin
PMM−X(14-X)
010
(n-2)*10
20
50
n * 10
R
50 Chicago
60 Cincinnati
ID=10
0 Atlanta
1 Austin
n-2 Chicago
n-1 Cincinnati
Division point n-2

FIG. 19

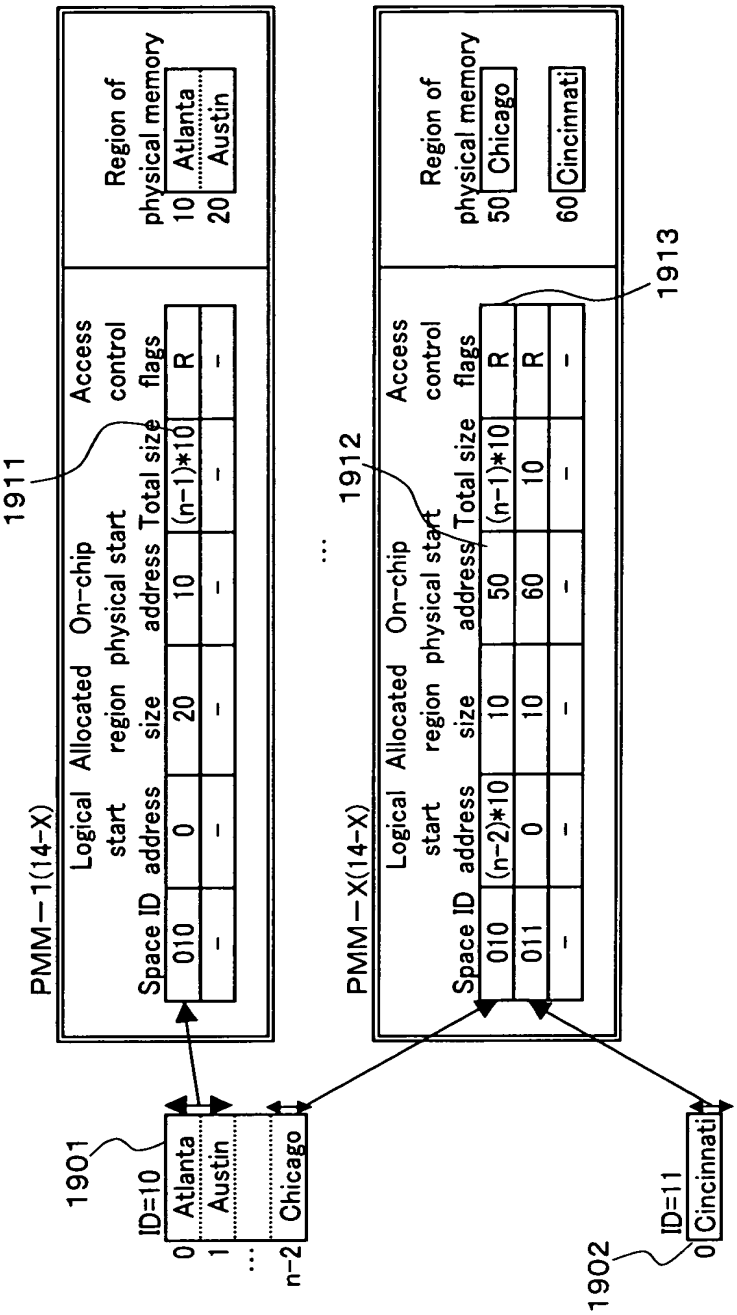

PMM−1(14-X)
Space ID
Logical start address
Allocated region size
On-chip physical start address
Total size
Access control flags
010
0
20
10
(n−1)*10
R
Region of physical memory
10 Atlanta
20 Austin
1911
PMM−X(14-X)
Space ID
Logical start address
Allocated region size
On-chip physical start address
Total size
Access control flags
010
(n−2)*10
10
50
(n−1)*10
R
011
0
10
60
10
R
Region of physical memory
50 Chicago
60 Cincinnati
1912
1913
1901
ID=10
0 Atlanta
1 Austin
n−2 Chicago
1902
ID=11
0 Cincinnati

FIG. 20

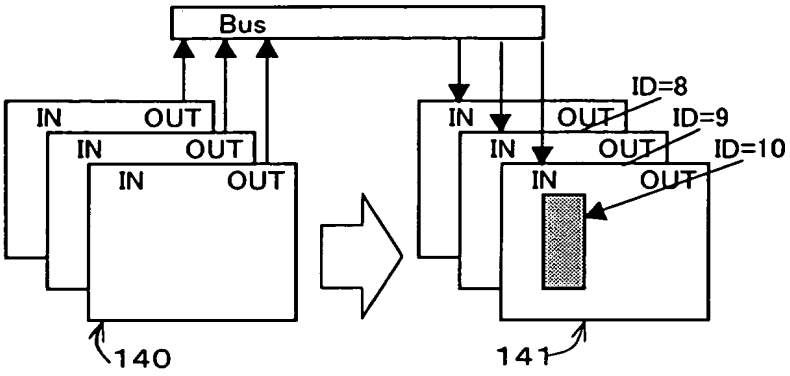

Bus
IN
OUT
IN
OUT
IN
OUT
140
ID=8
ID=9
ID=10
IN
OUT
IN
OUT
IN
OUT
141

FIG. 21

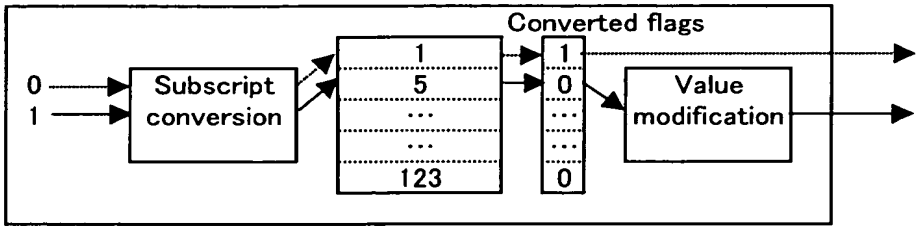

Converted flags
0
1
Subscript conversion
1
5
...
...
123
1
0
...
...
0
Value modification

FIG. 22

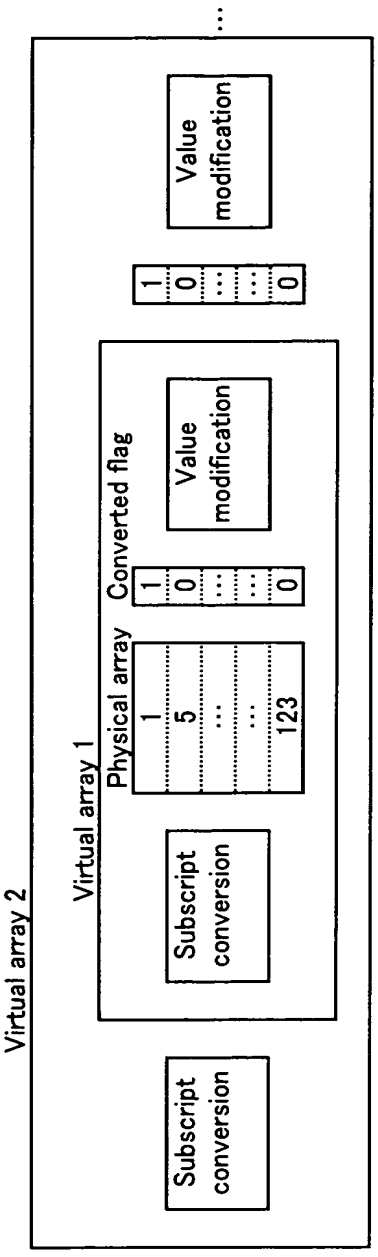

Virtual array 2
Subscript conversion
Virtual array 1
Subscript conversion
Physical array
1
5
...
...
123
Converted flag
1
0
...
...
0
Value modification
1
0
...
...
0
Value modification
...

FIG. 23

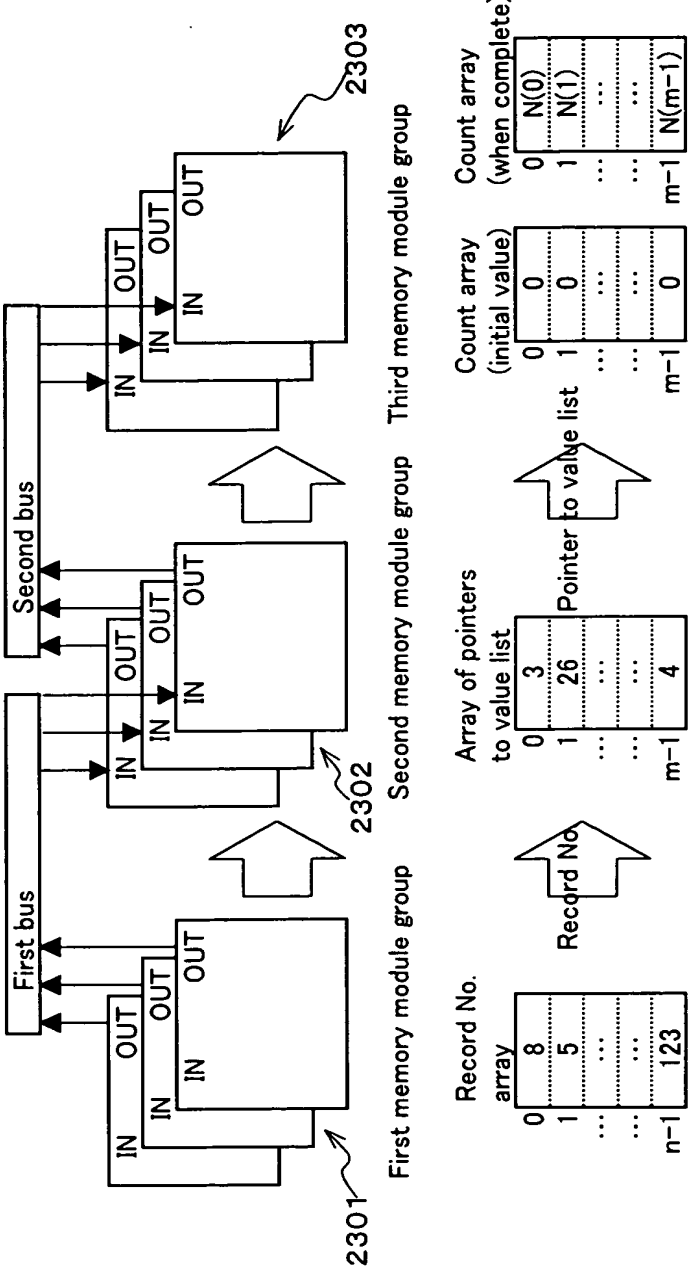

First bus
Second bus
IN
OUT
2301
2302
2303
First memory module group
Second memory module group
Third memory module group
Record No. array
Record No
Array of pointers to value list
Pointer to value list
Count array (initial value)
Count array (when complete)
0
1
n-1
m-1
8
5
123
3
26
4
N(0)
N(1)
N(m-1)

FIG. 24

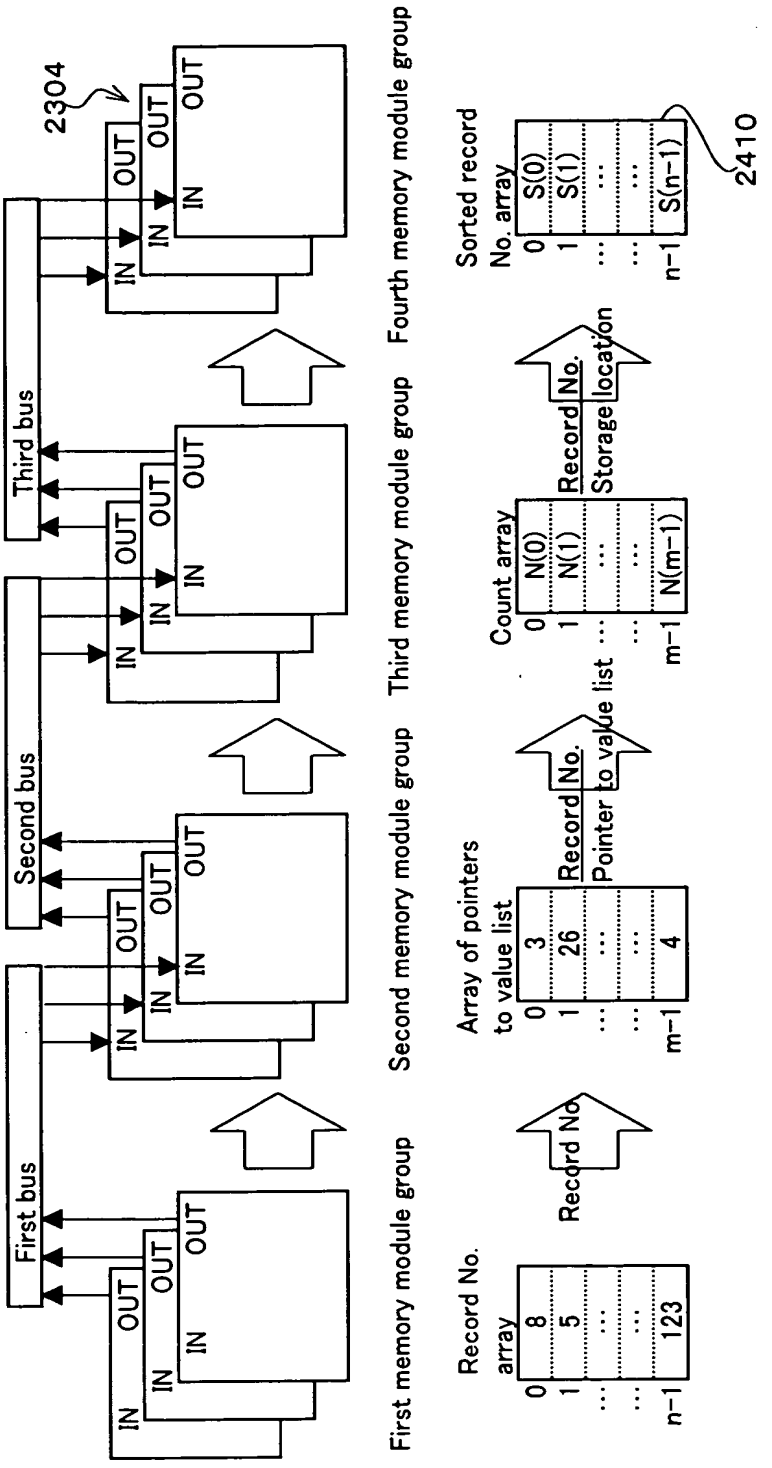

First bus
Second bus
Third bus
IN
OUT
2304
First memory module group
Second memory module group
Third memory module group
Fourth memory module group
Record No. array
0
1
n-1
8
5
123
Record No
Array of pointers to value list
0
1
m-1
3
26
4
Record No.
Pointer to value list
Count array
N(0)
N(1)
N(m-1)
Record No.
Storage location
Sorted record No. array
S(0)
S(1)
S(n-1)
2410

FIG. 25

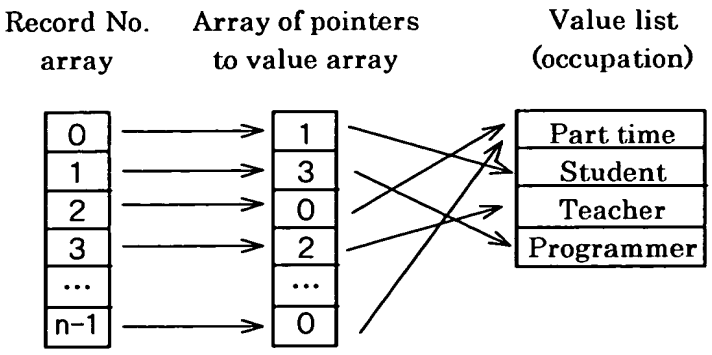

Record No. array
Array of pointers to value array
Value list (occupation)
0
1
2
3
...
n-1
1
3
0
2
...
0
Part time
Student
Teacher
Programmer

FIG. 27

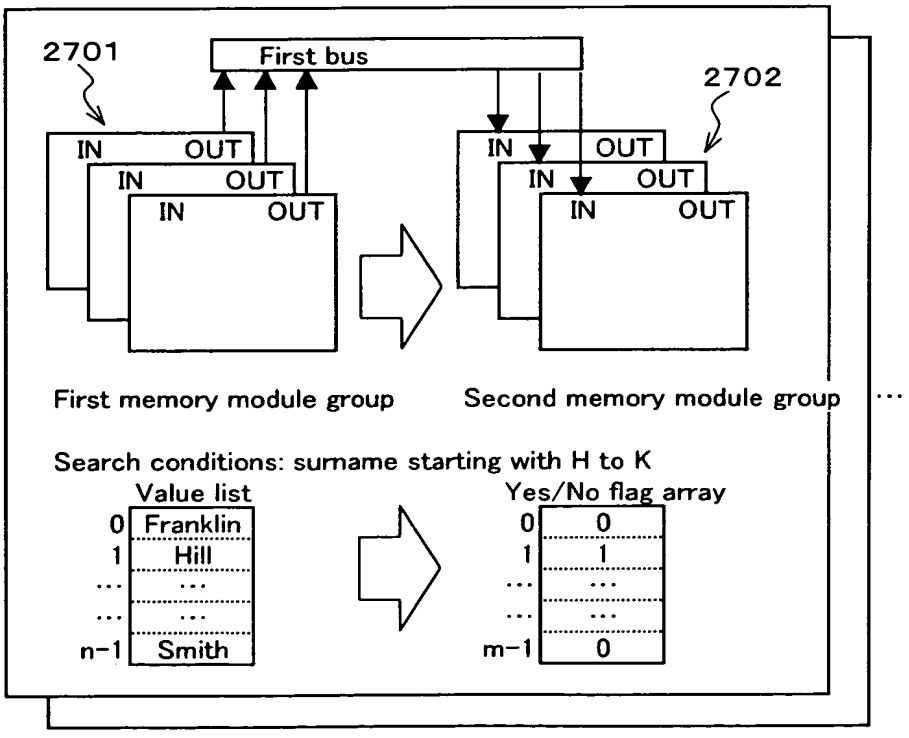

2701
First bus
2702
IN
OUT
First memory module group
Second memory module group
Search conditions: surname starting with H to K
Value list
Yes/No flag array
0
Franklin
1
Hill
n-1
Smith
m-1

FIG. 26

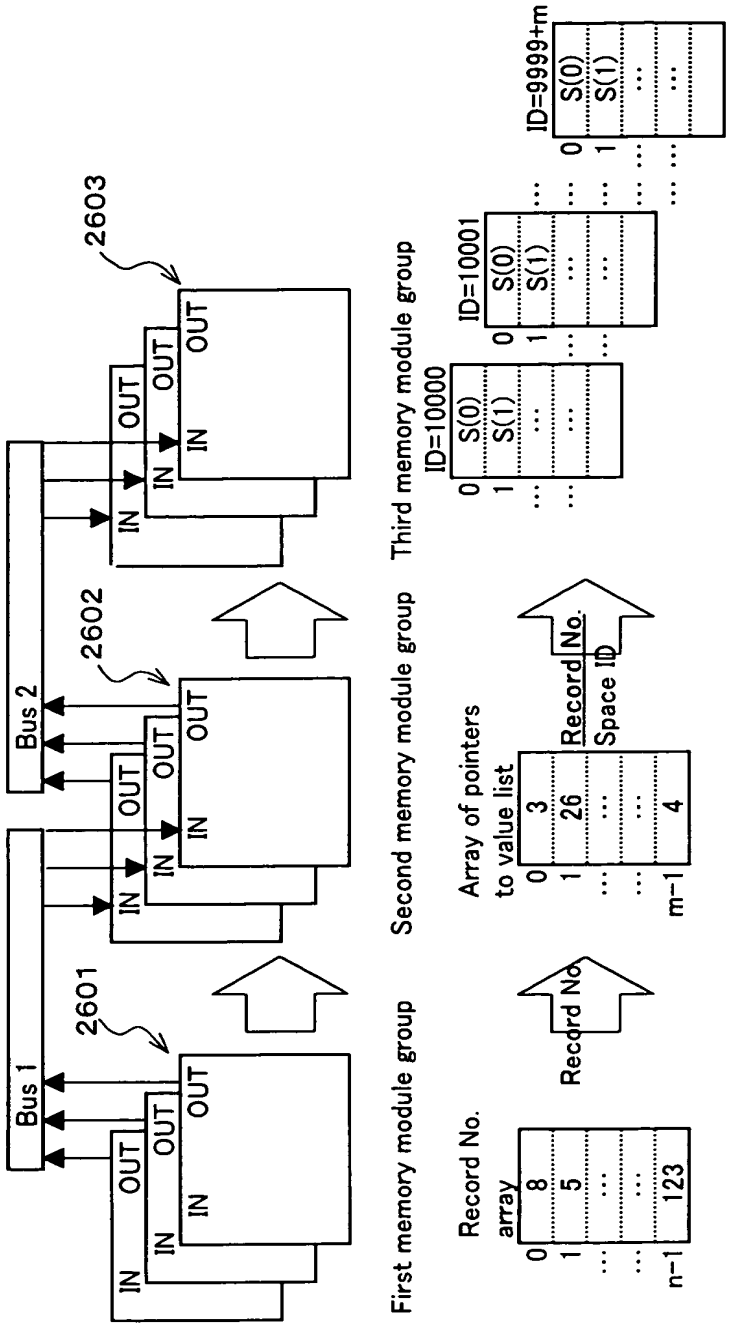
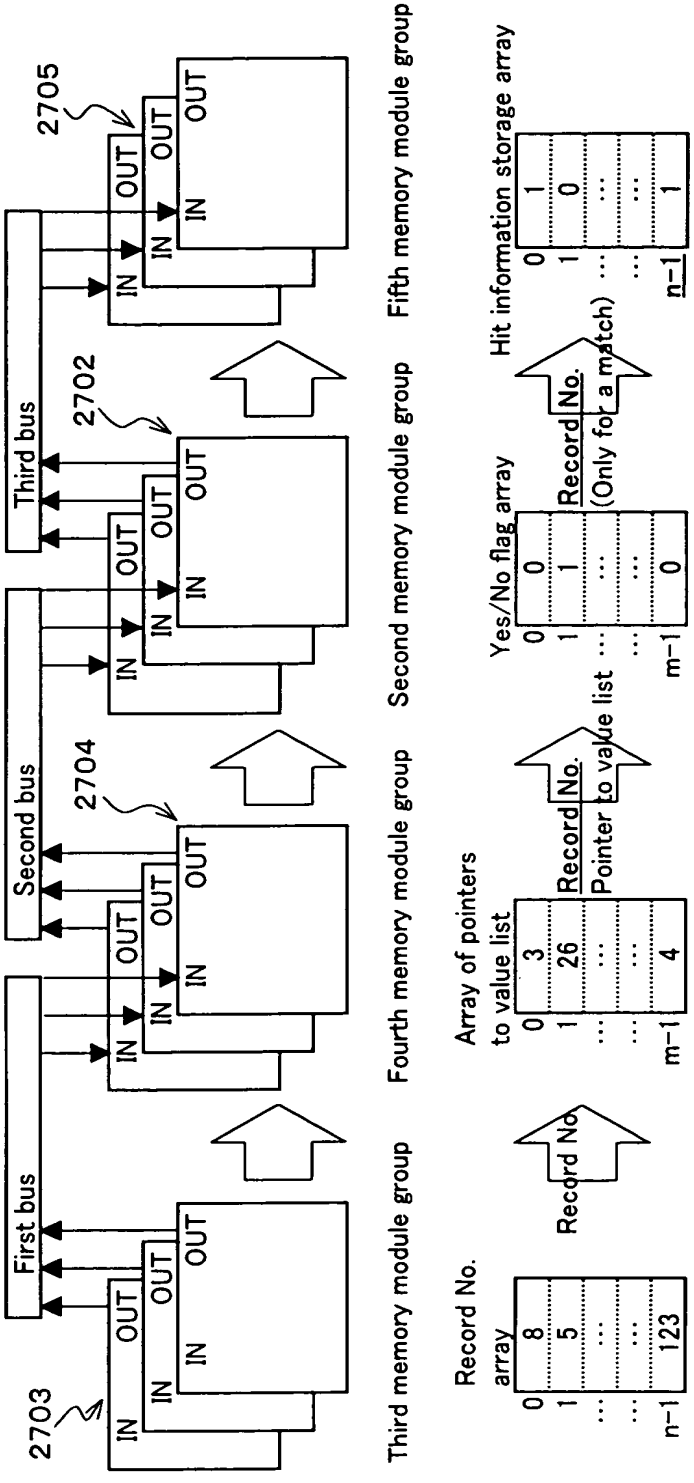

Bus 1
Bus 2
2601
2602
2603
IN
OUT
First memory module group
Second memory module group
Third memory module group
Record No. array
Record No
Array of pointers to value list
Record No.
Space ID
ID=10000
ID=10001
ID=9999+m
S(0)
S(1)

FIG. 28


2703
2704
2702
2705
First bus
Second bus
Third bus
IN
OUT
Third memory module group
Fourth memory module group
Second memory module group
Fifth memory module group
Record No. array
Record No
Array of pointers to value list
Record No.
Pointer to value list
Yes/No flag array
Record No.
(Only for a match)
Hit information storage array
0
1
m-1
n-1
8
5
123
3
26
4
0
1
0
1